

## A Proposed Algorithm for Image Enhancement using DMWT

Shaimaa A. Fahdel

Department of Computer Science, College of Education, Ibn-Al-Haithen, Baghdad University,  
Baghdad-Iraq.

**Abstract**

This paper proposes a method of color image enhancement using cascaded application of multiwavelet transform. Our enhancement contains two fields one for remove or decrease the amount of noise from noisy image while the other field deals with applying mean filter for give the resultant image more soft appearance and to make the image more smoother. Also, it suggests a method for computing the multiwavelet transform using the 1<sup>st</sup> order approximation. The proposed method of the enhancement showed an acceptable result in comparison with the conventional methods that are used for image enhancement. The proposed algorithm and the evaluation test carried using Delphi V.5 package.

**Introduction:-**

Image enhancement is an important field in digital image processing. Transforms play a great role in this field since they offer a new representation for the data in which features to be processed become more distinct [1]. This research presents how to enhance color image using Multiwavelet transformation mixed with the mean filter to get a new enhanced color image.

**Multiwavelet Transform**

Two methods can be used for computing of multiwavelet transform with two-dimensional signals. They are [2]; Separated algorithm, and un-Separated algorithm.

Separated algorithm is working sequentially on each dimension by processing each row and then each column separately of the image. While the un-Separated algorithm method is working in both image dimensions at the same time. Regardless of its computational complexity, un-Separated algorithm can offer benefits over Separated algorithm. By using the low-pass filter ( $H_i$ ) and high-pass filter ( $G_i$ ), the transformation matrix can be created as shown [1, 3, 4]:-

$$\begin{bmatrix} H_0 & H_1 & H_2 & H_3 & 0 & 0 \\ G_0 & G_1 & G_2 & G_3 & 0 & 0 \\ 0 & 0 & H_0 & H_1 & H_2 & H_3 \\ 0 & 0 & G_0 & G_1 & G_2 & G_3 \end{bmatrix} \dots\dots\dots(1)$$

$H_i$  and  $G_i$  is  $2 \times 2$  matrices having the following values shown in table no. 1.

**Table (1)**  
**Values of low-pass filter ( $H_i$ ) and high-pass filter ( $G_i$ ).**

	R. (0)	C. (0)	R. (0)	C. (1)	R. (1)	C. (0)	R. (1)	C. (1)
$H_0$	$\frac{3}{5\sqrt{2}}$	$\frac{4}{5}$	$\frac{-1}{20}$	$\frac{-3}{10\sqrt{2}}$				
$H_1$	$\frac{3}{5\sqrt{2}}$	0	$\frac{9}{20}$	$\frac{1}{\sqrt{2}}$				
$H_2$	0	0	$\frac{9}{20}$	$\frac{-3}{10\sqrt{2}}$				
$H_3$	0	0	$\frac{-1}{20}$	0				
$G_0$	$\frac{-1}{20}$	$\frac{3}{10\sqrt{2}}$	$\frac{1}{10\sqrt{2}}$	$\frac{3}{10}$				
$G_1$	$\frac{9}{20}$	$\frac{-1}{\sqrt{2}}$	$\frac{9}{10\sqrt{2}}$	0				
$G_2$	$\frac{9}{20}$	0	$\frac{9}{10\sqrt{2}}$	$\frac{3}{10}$				
$G_3$	$\frac{1}{20}$	0	$\frac{-1}{10\sqrt{2}}$	0				

**DMWT Computation Algorithm**

In this method of preprocessing when using the approximation-based scheme of preprocessing, the input image must be  $N \times N$  dimension and  $N$  must be power of 2 also the DMWT matrix should have the same dimensions of the input image. If this

condition is not found , we can solve this problem by adding rows or columns of values equal to zero to get a square matrix [5, 6, 7]:-

1. The dimension of Image matrix is a square matrix  $N \times N$ , as explained previously.
2. Create the transformation matrix depending on equation (1), with the values of low-pass and high- pass filter as shown in Table (1).

Note that the transformation matrix of dimension  $(N/2 \times N/2)$  will be expanded and get the dimension  $(N \times N)$  after substituting GHM matrix filter coefficients values as given in Table (1) each value in that transformation matrix will get it's four values by replacing with its  $(2 \times 2)$  matrix, so the dimension of this matrix will be  $(8 \times 8)$ :-

$$\begin{bmatrix} H_{0,0,0} & H_{0,0,1} & H_{1,0,0} & H_{1,0,1} & H_{2,0,0} & H_{2,0,1} & H_{3,0,0} & H_{3,0,1} \\ H_{0,1,0} & H_{0,1,1} & H_{1,1,0} & H_{1,1,1} & H_{2,1,0} & H_{2,1,1} & H_{3,1,0} & H_{3,1,1} \\ G_{0,0,0} & G_{0,0,1} & G_{1,0,0} & G_{1,0,1} & G_{2,0,0} & G_{2,0,1} & G_{3,0,0} & G_{3,0,1} \\ G_{0,1,0} & G_{0,1,1} & G_{1,1,0} & G_{1,1,1} & G_{2,1,0} & G_{2,1,1} & G_{3,1,0} & G_{3,1,1} \\ H_{2,0,0} & H_{2,0,1} & H_{3,0,0} & H_{3,0,1} & H_{0,0,0} & H_{0,0,1} & H_{1,0,0} & H_{1,0,1} \\ H_{2,1,0} & H_{2,1,1} & H_{3,1,0} & H_{3,1,1} & H_{0,1,0} & H_{0,1,1} & H_{1,1,0} & H_{1,1,1} \\ G_{2,0,0} & G_{2,0,1} & G_{3,0,0} & G_{3,0,1} & G_{0,0,0} & G_{0,0,1} & G_{1,0,0} & G_{1,0,1} \\ G_{2,1,0} & G_{2,1,1} & G_{3,1,0} & G_{3,1,1} & G_{0,1,0} & G_{0,1,1} & G_{1,1,0} & G_{1,1,1} \end{bmatrix}$$

3. Processing the odd and even rows of the input image using the following equations [5, 6]:-

For any Odd-Row:-

$$\text{New Odd-Row} = A_1 \times [\text{current odd row}] + A_2 \times [\text{next even row} + \text{previous even row}] \dots\dots\dots(2)$$

Where:-  $A_1 = 0.373615$   
 $A_2 = 0.11086198 \dots\dots\dots(3)$

For any Even Row:-New even-row= $(\sqrt{2}-1)$  [current even-row]

If we compute the first odd row, the value of previous even-row in equation (2) equals to zero and the same manner for computing the last odd row the value of the next even-row equals to zero.

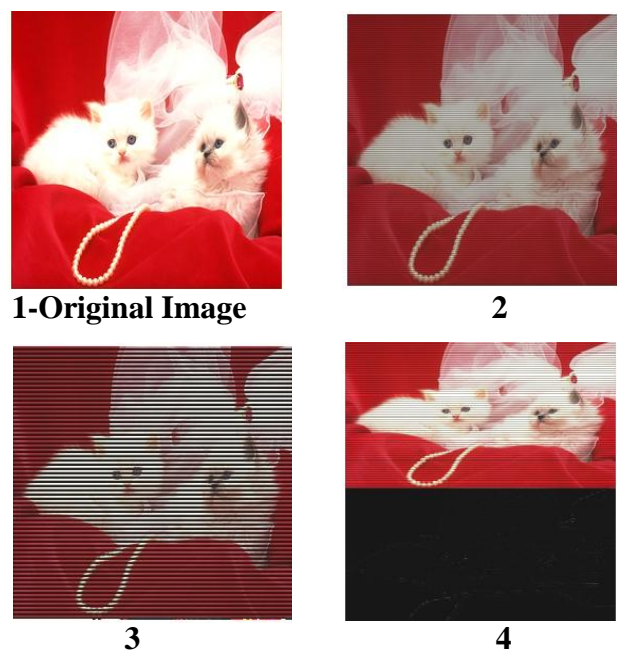
4. Multiply to the transformation matrix in (1) by the  $N \times N$  resulted image matrix from the previous step then rearrange rows by dividing the resulted matrix into two equal

half and putting the rows number 1,2 and 5,6 ...  $N-3,N-2$  after each other in the upper-half and the rows number 3,4 and 7,8 ...  $N-1,N$  after each other in the next lower half. Note that this step will represent how to applying the transformation processing on the image.

5. Transpose the row of the resulted matrix from step 4 then repeats step 3.
6. Multiply to the transformation matrix in (1) by the  $N \times N$  resulted image matrix from the previous step then rearrange rows by dividing the resulted matrix into two equal half and putting the rows number 1, 2 and 5, 6 ...  $N-3, N-2$  after each other in the upper-half and the rows number 3, 4 and 7, 8 ...  $N-1, N$  after each other in the next lower half. Note that this step will represent how to applying the transformation processing on the image.
7. The Final output Matrix is the transpose of the resultant matrix from step 6 then divide the matrix into four sub-bands with arranging the rows in the same manner in steps 4 and 6.

### A Computer Implementation

A general computer program computing a single-level DMWT using a Critical- Sampled Scheme of Preprocessing (1<sup>st</sup> order approximation) are written using **DELPHI V.5.0** for a general  $N \times N$  image as shown in Fig. (1) step by step.



1-Original Image

2

3

4



**10- Final DMWT**  
**Fig. (1) Step by step of applying DMWT using a Critical-Sampled Scheme of Preprocessing (1<sup>st</sup> order approximation).**

**Image filtering with Mean Filter**

The meaning of image filtering is that you have a 2D filter matrix, and the 2D image. Then, for every pixel of the image, take the sum of products. Each product is the color value of the current pixel or a neighbor of it, with the corresponding value of the filter matrix. The center of the filter matrix has to be multiplied with the current pixel, the other elements of the filter matrix with corresponding neighbor pixels.

These points must be checked before applying filtering operation:

- The size of the filter must be odd to get a center point, so that it has a center, for example 3x3, 5x5 and so on.
- The summation of all elements of the filter must equal to 1 in case of wanting the resulted image having the same brightness as the original image. And in case the summation greater than 1, we can get a

resulted image brighter and vice versa. While the summation equal to 0, the resulted image will be very dark.

This research deals with Mean filter which is a linear filter used for give an image more smoother appearance by reducing the amount of variety of intensity between the neighbors pixels, also can be used to reduce the amount of noise in noisy image [5, 7]. The mean filtering is can be implemented simply to replace each pixel value in an image with the mean ('average') value of its neighbors. This filter can be implemented using convolution process by using a kernel of odd dimension. As shown below, a square kernel 3x3 is used for implementation

$$\begin{matrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{matrix}$$

The main description of how applying this convoluted filter can be thought as Putting the mask in the upper-left corner of the image then multiply the corresponding pixels, sum the results, then putting the result in the corresponding center pixel of the same [5, 6, 7] then moving the mask to the right side one pixel only and repeat the same processing to the end of row. When the row is ended the mask is moved down on pixel only with repeating the same processing.

For example if we have input image of size 5x5 and the kernel of size 3x3 as shown below:-

Image <sub>11</sub>	Image <sub>12</sub>	Image <sub>13</sub>	Image <sub>14</sub>	Image <sub>15</sub>
Image <sub>21</sub>	Image <sub>22</sub>	Image <sub>23</sub>	Image <sub>24</sub>	Image <sub>25</sub>
Image <sub>31</sub>	Image <sub>32</sub>	Image <sub>33</sub>	Image <sub>34</sub>	Image <sub>35</sub>
Image <sub>41</sub>	Image <sub>42</sub>	Image <sub>43</sub>	Image <sub>44</sub>	Image <sub>45</sub>
Image <sub>51</sub>	Image <sub>52</sub>	Image <sub>53</sub>	Image <sub>54</sub>	Image <sub>55</sub>

Kernal <sub>11</sub>	Kernal <sub>1</sub> 2	Kernal <sub>13</sub>
Kernal <sub>21</sub>	Kernal <sub>2</sub> 2	Kernal <sub>23</sub>
Kernal <sub>31</sub>	Kernal <sub>3</sub> 2	Kernal <sub>33</sub>

To implement the convolution process the following equation is used:-

$$\text{OUTPUT IMAGE } (i, j) = \sum_{d=1}^m \sum_{f=1}^n I(i+d-1, j+f-1) \times K(d, f) \dots\dots\dots(4)$$

Where *I* represents the input image and *K* represents the Kernel value. So the implementation of the filter as follows:-

$$\begin{aligned} \text{Image}_{22} = & (\text{Image}_{11} \times \text{Kernal}_{11}) + (\text{Image}_{12} \times \text{Kernal}_{12}) + \\ & (\text{Image}_{13} \times \text{Kernal}_{13}) + (\text{Image}_{21} \times \text{Kernal}_{21}) + \\ & (\text{Image}_{22} \times \text{Kernal}_{22}) + (\text{Image}_{23} \times \text{Kernal}_{23}) + \\ & (\text{Image}_{31} \times \text{Kernal}_{31}) + (\text{Image}_{32} \times \text{Kernal}_{32}) + \\ & (\text{Image}_{33} \times \text{Kernal}_{33}). \end{aligned}$$

**A proposed algorithm for improvement of image enhancement using DMWT with mean filter**

In this section, new method is proposed for color image enhancement using the mean filter which is applied two times on some parts of DMWT transformed image, as shown in the Fig.(2).

The steps of the proposed method are as follows:-

1. Obtain the multiwavelet transformation coefficients of the observed input color image.
2. Apply enhancement approach on some parts of the resultant image by applying mean filter on the high pass components only (LH ( $L_1H_1, L_1H_2, L_2H_1, L_2H_2$ ), HL( $H_1L_1, H_1L_2, H_2L_1, H_2L_2$ ), and HH( $H_1H_1, H_1H_2, H_2H_1, H_2H_2$ )) and leave the low-pass components as it is which is used in the next step.

$L_1L_1$	$L_1L_2$	$LH$
$L_2L_1$	$L_2L_2$	
$HL$		$HH$

3. Obtain the Multiwavelet Transformation coefficients for each part of the low-pass components *LL* ( $L_1L_1, L_1L_2, L_2L_1, L_2L_2$ )

then repeat step (2) on the resultant high-pass components only (LH ( $L_1H_1, L_1H_2, L_2H_1, L_2H_2$ ), HL( $H_1L_1, H_1L_2, H_2L_1, H_2L_2$ ), and HH( $H_1H_1, H_1H_2, H_2H_1, H_2H_2$ )) by applying mean filter on that components and leave the low-pass components.

4. Inverse the multiwavelet transformation coefficients of the resultant image from the previous step two times since the multiwavelet transform is applied two times so we must apply the inverse Multiwavelet transform two times to get the reconstructed image.

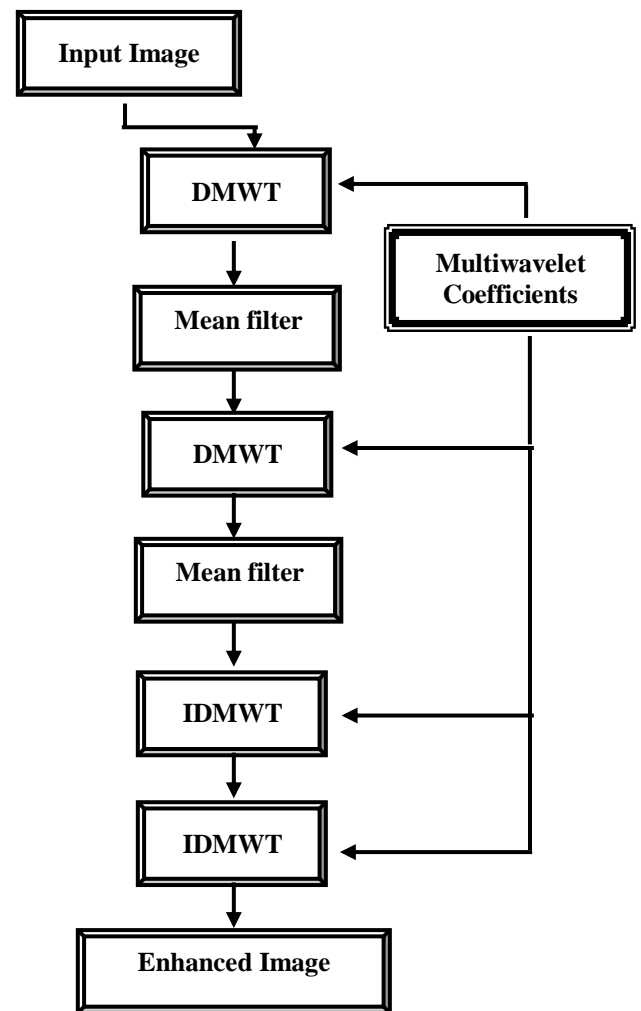


Fig. (2):- A flow diagram of this method.

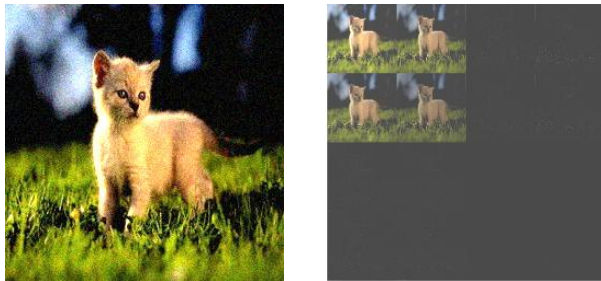
**System Implementation**

The goal of this system is to enhance a color image depending on the multiwavelet transform with smoothing filter (mean filter).

The system is designed by dividing into two main parts:-

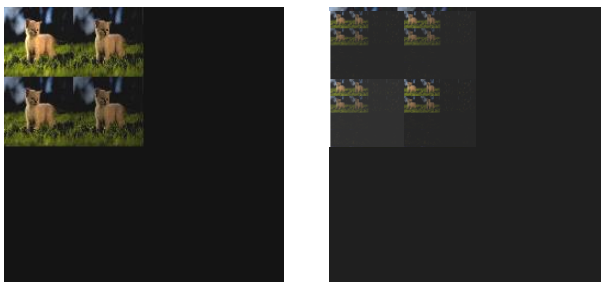
1. Multiwavelet transforms using a Critical-Sampled Scheme of Preprocessing (1<sup>st</sup> order approximation) and its inverse, which is applied two times
2. Enhancement Mean filter which is applied two times on LL, HL and HH bands.

Fig. (3 and 4) show the full implementation of this algorithm.



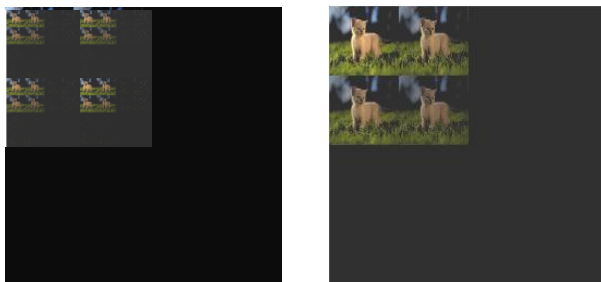
1- Original input Image.

2- Applying DMWT transformation.



3- Applying smoothing on High frequency.

4- Applying DMWT on each part of the LL bands.



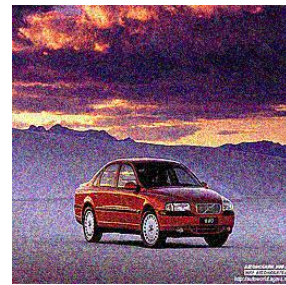
5- Applying smoothing on the four sub-bands of the resultant High frequency.

6- Applying Inverse DMWT to the four sub-bands.

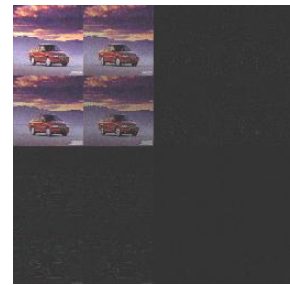


7- Reconstructed Image after applying inverse DMWT on the resultant image from previous step.

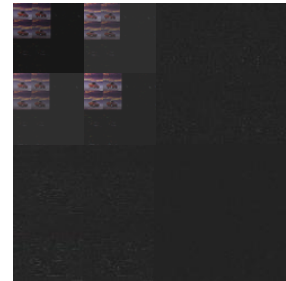
**Fig. (3) Applying the steps of algorithm on an enhanced CAT image.**



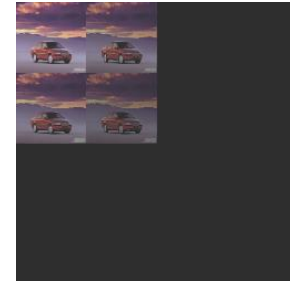
1- Noisy Image (Original Image).



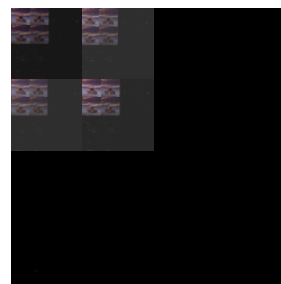
2- Applying DMWT transformation.



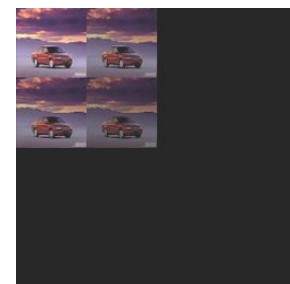
4- Applying DMWT on each part of the LL bands.



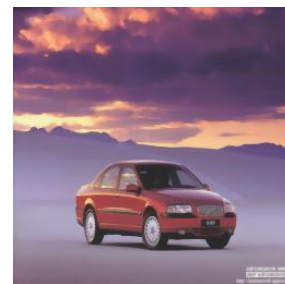
3- Applying smoothing on LH, HL and HH bands.



5- Applying smoothing on the four sub-bands of the resultant High frequency.



6- Applying Inverse DMWT to the four sub-bands.



7- Reconstructed Image after applying inverse DMWT on the resultant image from previous step.

**Fig. (4) Applying the steps of algorithm on Noisy RED-CAR image with amount of noise.**

Here is the table of values for comparing between the initial un-enhanced image and the resultant final enhanced image to improve the efficiency of the proposed algorithm using SNR formula as mentioned below:-

$$SNR = 10 \log_{10} \left[ \frac{\sum_{r=0}^{M-1} \sum_{c=0}^{N-1} (\hat{i}(r, c))^2}{\sum_{r=0}^{M-1} \sum_{c=0}^{N-1} (X(r, c) - (r, c))^2} \right] \dots\dots\dots(9)$$

Where:-

$\hat{I}(r, c)$  Represents the signal (the denoised image) with row size equal to M and Column size equal to N.

$X(r, c)$  Represent the errors (the noisy image) with row size equal to M and Column size equal to N.

Note that the large number for the value of SNR means we get a better image

**Table (2)**  
*Comparing between the results before and after applying the algorithm.*

	Initial Image1	Final Enhanced Image1	Initial Image2	Final Enhanced Image2
<b>Size</b>	H=256 W=256	H = 256 W = 256	H = 256 W =256	H = 256 W = 256
<b>SNR</b>	7.4401	10.0735	4.3087	12.2873

**Conclusions and Suggestions**

- The use of mean filter after the DMWT is improving the enhancement process.
- By computing the amount of SNR value before and after applying the adapted algorithm, one can conclude that the contrast between the results can documented as follows:-

a) For the first CAT image:-

$$C = SNR_{final} - SNR_{initial}$$

Where C is the contrast.

$$C = 10.0735 - 7.4401 = 2.6334 \text{ (two times of the enhancement degree)}$$

b) For the second noisy RED-CAR image:-

$$C = 12.2873 - 4.3087 = 7.9786 \text{ (seven times of the denoising degree).}$$

That means the advantage of this algorithm is to get greater value for SNR which led to get better image when it is applied on noisy images, since it can gain a large amount of SNR as well as can be applied for increasing the degree of enhancement depending on the type of the filter used with the DMWT.

- Multiwavelet transform plays an important role in image enhancement due to its multi-resolution analysis that makes the statistics of many natural images be simplified when they are decomposed. It takes the idea of wavelet transform with different two-dimensional filters and with pre/post processing mechanisms. From the work in this research, the following points are concluded:-
  - The Multiwavelet can enhance the image better than using the wavelet transform.
  - Multiwavelet transform analyzes a signal at a nested set of scales. Hence, the multiwavelet transform matches both the long-duration and short-duration signal components.
- Trying to add the Multiwavelet transform for solving deblurring image processing problem.

**References**

[1] Mohammed, Z. J., “Video Image Compression Based on Multiwavelet Transforms”, University of Baghdad, Computer Science and Information System Dep. 2004.

[2] Y. LU, M. N. DO, “A Directional Extension for Multidimensional Wavelet Transforms”, EEE Transactions on Image Processing, submitted; 2005.

[3] Hadi, A. H., 2004, “Wavelet-Domain Reconstruction of Lost Blocks in Image” M.Sc. Thesis, University of Technology, Baghdad.

[4] Saleh, Z. J. M., 2004, “Image Video Compression Using Multi-Wavelet Transform”, Ph.D Thesis, University of Baghdad, Baghdad.

[5] Rafael C. Gonzalez and Richard E. Woods “Digital Image Processing”, 2001.

[6] Strela, V., “Multiwavelet: Theory and Applications Transforms”, Ph.D. thesis, Dep. Of Math., Massachusetts Institute of Technology, June 1996.

[7] Wang Ling, “Orthogonal Multiwavelet Transform for Image Denoising”, Proceedings of ICSP2000, pp. 987-991, 2000.

### الخلاصة

يقترح هذا البحث طريقة لتحسين الصور الملونة من خلال استخدام عدد من تطبيقات تحويل متعدد المويجة. نوعية التحسين المستخدمة في هذا البحث تتضمن جانبين اساسيين الاول يهتم بازالة او تقليل كمية التشويه الموجودة في الصور قدر الامكان بينما يهتم الجانب الاخر بتطبيق مرشح المعدل من اجل التقليل من حدة الصورة الناتجة من تطبيق الجانب الاول.

ايضاً يقترح هذا البحث طريقة لكيفية حساب نوع من انواع تحويلات متعدد المويجة والذّي يعرّف بـ (1<sup>st</sup> order approximation).

الطريقة المقترحة في عملية التحسين اظهرت نتائج مقبولة بالمقارنة مع الطرق التقليدية المستخدمة في عمليات تحسين الصور الملونة. تم تقييم الخوارزمية التي اقترحت في هذا البحث من خلال استخدام لغة باسكال المرئية (Delphi V.5 Package) ذو الاصدار الخامس.